

# Pneumatic Reservoir Computing for Sensing Soft Body: Computational Ability of Air in Tube and Its Application to Posture Estimation of Soft Exoskeleton

Toshihiro Kawase,<sup>1,2\*</sup> Tetsuro Miyazaki,<sup>3</sup> Takahiro Kanno,<sup>4</sup>  
Kotaro Tadano,<sup>2</sup> Yoshikazu Nakajima,<sup>1</sup> and Kenji Kawashima<sup>3</sup>

<sup>1</sup>Institute of Biomaterials and Bioengineering, Tokyo Medical and Dental University,  
2-3-10 Kanda-Surugadai, Chiyoda, Tokyo 101-0062, Japan

<sup>2</sup>Institute of Innovative Research, Tokyo Institute of Technology,  
4259 Nagatsuta-cho, Midori-ku, Yokohama, Kanagawa 226-8503, Japan

<sup>3</sup>Graduate School of Information Science and Technology, The University of Tokyo,  
Yushima 1-5-45, Bunkyo, Tokyo 113-8510, Japan

<sup>4</sup>Riverfield, Inc., Yotsuya Medical Bldg. 5th floor, 20 Samon-cho, Shinjuku, Tokyo 160-0017, Japan

(Received February 25, 2021; accepted June 29, 2021)

**Keywords:** soft robotics, physical reservoir computing, pneumatic system, pipelines, posture estimation

Sensing and computing the body states of soft-bodied robots require new methods. Recent studies on soft robotics have shown their feasibility to be applied for these purposes; however, they only addressed solid parts and not the behavior of inner fluids in soft robotics. In this study, we investigated the possibility of a framework that can be used to estimate the body state by exploiting air dynamics in tubes connecting chambers. The framework was designed on the basis of the concept of physical reservoir computing. We focused on a case with a single tube connection. A benchmark task emulated a nonlinear system and was evaluated by simulation. The results showed that the computational ability depended on the inner diameter and length of the tube and can be increased by selecting a suitable diameter and length. We physically implemented the framework for the posture estimation of a soft exoskeleton using pneumatic artificial rubber muscles (PARMs) as the connected chambers and evaluated the accuracy of estimation of a thigh angle. The estimation accuracy showed a similar trend as a function of the tube properties as that observed in the simulation. The framework can exploit the dynamics of air in a tube and may be useful for the state estimation of soft-bodied robots.

## 1. Introduction

Sensing and computation are essential in controlling robots. In most cases of the control of robots, sensor signals are transformed to a state description on the robot body (e.g., joint angles) and exploited for control. The transformation is usually processed by computer software based on the models of the robot bodies, which have conventionally consisted of rigid bodies. Nowadays, robotics also covers soft-bodied robots, which are mainly made from soft materials.

---

\*Corresponding author: e-mail: kawase.bmi@tmd.ac.jp  
<https://doi.org/10.18494/SAM.2021.3345>

In many cases, soft robots are difficult to model in a mathematical form. Therefore, new algorithms or methods for sensing and computation are required.<sup>(1)</sup>

Recent studies have shown that the physical body of a robot itself can be used for computation and applied to the sensing and control of soft-bodied robots.<sup>(2)</sup> Some studies have shown that the complex dynamics of soft materials provides rich resources for computation. In the control of some robots, locomotion was controlled by using signals from sensors embedded in the soft part of the body.<sup>(3,4)</sup> It has been shown that a soft arm resembling an octopus arm can be exploited for some computational tasks, such as a parity check.<sup>(5,6)</sup> These methods, however, use some movements of the robot itself for computation and require a specially designed body morphology [e.g., tensegrity structure<sup>(4)</sup>] to be used in combination with the robot's primarily required motion (e.g., locomotion).

In the research of soft-bodied robots, the softness of solid materials has mainly been considered. Here, we focus on air surrounded by a material. We hypothesize that the dynamics of air can be exploited as a computational resource for soft robots. The flow and compressibility of air are expected to exhibit more complex behaviors than solids, which may be useful for obtaining higher computational power in soft robots. Although some studies have assessed the computational ability of two-dimensional fields of fluids,<sup>(7,8)</sup> a hard stage was required to observe flow propagation in the two-dimensional space, which is difficult to implement in a soft robot. Frameworks that can be implemented with soft materials should be investigated for soft robot applications.

As a physical computation method that can be incorporated into soft robots without specially designed body shapes, we propose a method of sensing and computation that uses the dynamics in air in chambers connected by tubes [Fig. 1(c)]. In this method, deformable air chambers connected by tubes are embedded in a robot. The chambers transmit the deformation of the robot body to the internal air. The changes in pressure propagate in the tubes and are measured by sensors. On the basis of the framework of physical reservoir computing,<sup>(9)</sup> the sensor signals are multiplied with pre-optimized weights, and the sum of the values is used as the output. With flexible tubes, the system can be embedded in soft-bodied robots without the loss of softness. Unlike existing studies of physical reservoir computing with manipulators using pneumatic soft actuators,<sup>(10,11)</sup> the proposed method does not rely on the dynamics of the solid body of the robot. This feature separates the function of computation from the movement of the robot and is expected to contribute to the simpler design of robots performing physical computation.

In this paper, we investigated the computational ability of such a system in the simplest case with only one tube connection. In the following sections, we first illustrate the concept of physical reservoir computing and its application to air dynamics. Next, we investigate the computational ability of a tube by simulation. We model a simple essential system, which has a chamber and a tube, and show that computational ability can be embedded in the system by using appropriate parameters of the tube (inner diameter and length). Finally, we show an application of the method to sensing the joint angle of a soft exoskeleton using two pneumatic artificial muscles as chambers and a tube connection.

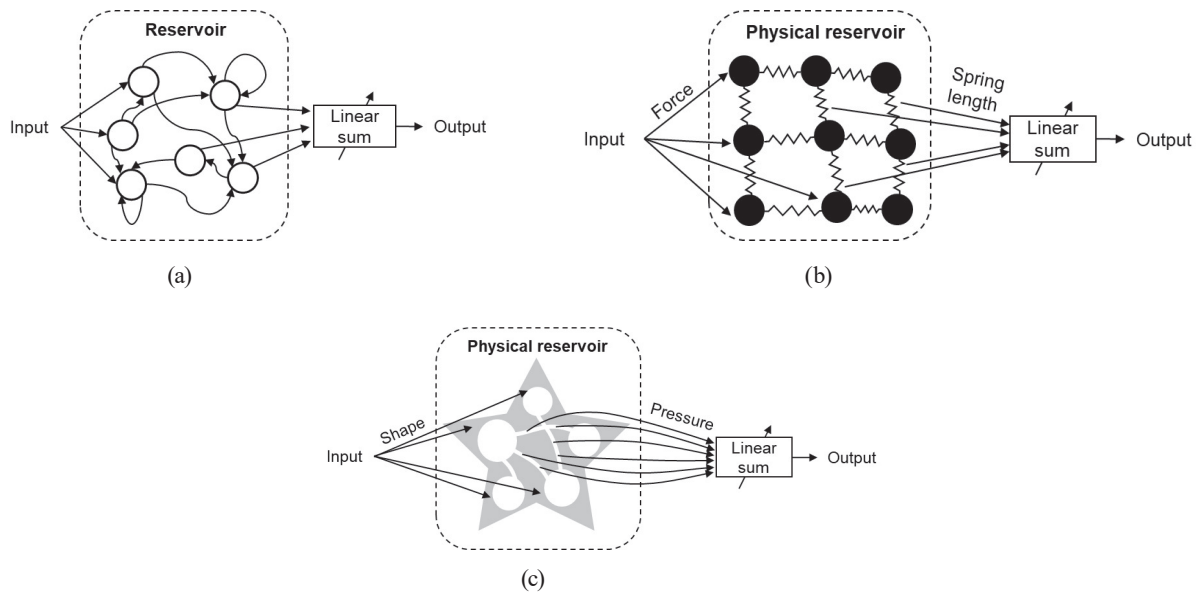


Fig. 1. (a) Reservoir computing,<sup>(12,13)</sup> (b) physical reservoir computing using a mass-spring system,<sup>(14)</sup> and (c) proposed physical reservoir computing using a pneumatic pipeline system.

## 2. Concept of Proposed Method

Physical reservoir computing is a method derived from the supervised machine learning method of reservoir computing. Reservoir computing is a neural network method originally known as an echo state network<sup>(12)</sup> or liquid state machine.<sup>(13)</sup> In this method, a randomly coupled neural network, called a reservoir, is used to create various signals that reflect the input, and the desired output is obtained as a linear sum [Fig. 1(a)]. The weights of the randomly connected network are fixed, and only the weights of the linear sums are adjusted in training by ordinary least squares, ridge regression, and so forth. This makes the computational cost of learning very low.

Physical reservoir computing is a method in which the fixed network is replaced by a physical system that shows complex behavior, called a physical reservoir [Fig. 1(b)]. A physical reservoir for computing requires high dimensionality, nonlinearity, fading memory, and a separation property.<sup>(9)</sup> High dimensionality is the ability to map inputs to high-dimensional physical variables used for the linear sum. Fading memory is memory in which the physical variables reflect recent but not all past inputs. Nonlinearity refers to the map from the current physical variables and inputs to the variables in the next step. The separation property means that the input signals that should be distinguished are mapped to different physical states. The dynamics of soft materials,<sup>(3–6)</sup> fluids,<sup>(7,8)</sup> mass-spring systems,<sup>(14,15)</sup> electronic circuits,<sup>(16)</sup> optical circuits,<sup>(17)</sup> memristors,<sup>(18)</sup> spintronic systems,<sup>(19)</sup> quantum systems,<sup>(20)</sup> and so forth have been exploited as computational resources for physical reservoir computing, and it has been experimentally shown that the systems have computational ability in benchmark tasks.

In our method, we use the air in chambers and pneumatic tubes as a physical reservoir [Fig. 1(c)]. The deformation caused by the robot’s motion changes the internal pressure of the chambers and generates complex behavior in the tubes, which is expected to be used as a computational resource for state estimation and control. As described in Sect. 3, the property of a pipeline gives a pressure distribution that has the required properties for physical reservoir computing. The computational ability of a pipeline is demonstrated by simulation in Sect. 3 in a benchmark task, and its ability to be used for state estimation is demonstrated in Sect. 4 in a soft robot application.

### 3. Computational Ability of Pipeline

In the proposed method, the property of the connection between the chambers is essential. If the chambers are connected by broad pathways, the pressure change in a chamber is instantly transmitted to the other chambers, and the state changes homogeneously. Instant transmission does not have fading memory, and the homogeneous state does not have high dimensionality or the separation property. As is known in pneumatics, transmission by a pipeline causes a delay and pressure loss.<sup>(21)</sup> The propagation of the pressure with a delay and loss will provide the fading memory property, and the emerging pressure distribution in the tube will have high dimensionality and the separation property.

In this paper, we focus on a case with a single tube. We modeled two chambers connected by a tube, with  $K$  pressure sensors measuring the pressure at points along the tube [Fig. 2(a)]. The first chamber is subjected to a force and deforms. The deformation causes pressure changes in the chamber. The pressure propagates in the pipeline and causes the changes of state in the

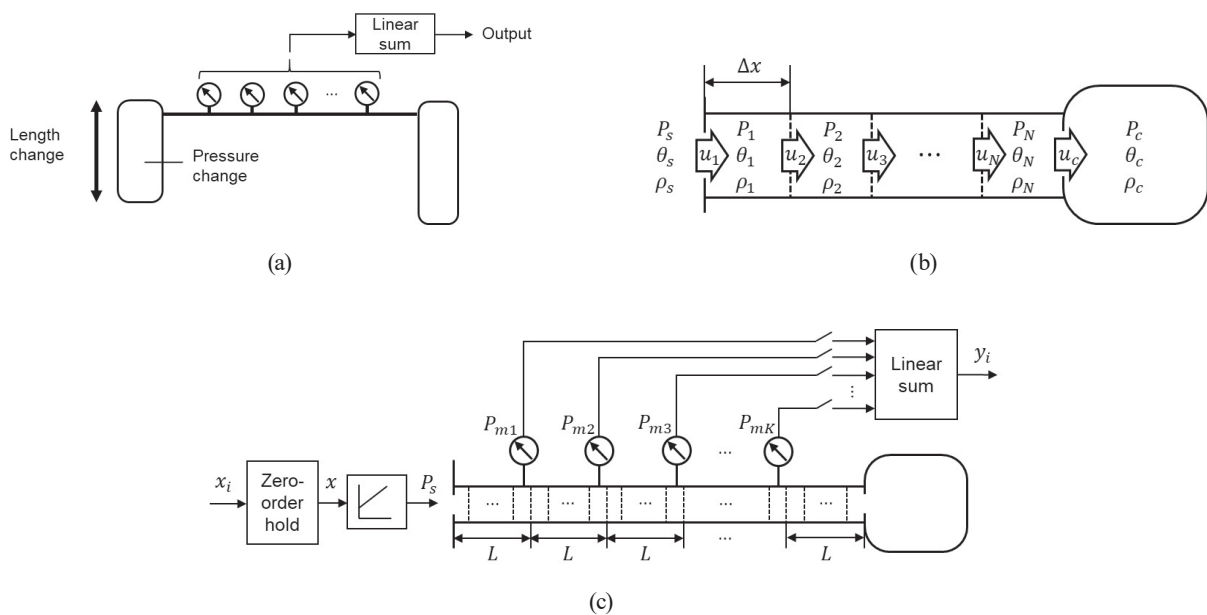


Fig. 2. Configuration of simulation. (a) Modeled situation, (b) grid model used for the simulation, and (c) experimental setup in the simulation.

second chamber. The values of the pressure sensors are used to calculate the output. Using the simulation model, we confirmed that the method has computational ability, and we investigated the properties of the pipeline that affect the computational ability.

### 3.1 Simulation model

To model the pressure propagation and distribution in the tube, we used a simulation method with a distributed model of a pipeline.<sup>(22,23)</sup>

#### 3.1.1 Input to the pipeline

The state change in the first chamber is assumed to be an isothermal process, that is,  $\theta_S = \text{const}$ . Pressure  $P_S$  is input to the system, which is assumed to be caused by deformation. The density  $\rho_S$  is changed according to the equation of state, that is,

$$\rho_S = \frac{P_S}{R\theta_S}, \quad (1)$$

where  $R$  is the gas constant.

The flow between the first chamber and the pipeline is modeled as an orifice flow. When the pressure on one side of an orifice  $P_A$  is greater than the pressure on the other side  $P_B$ , the air flows from the side with pressure  $P_A$  to the side with pressure  $P_B$ . The mass flow  $G(P_A, P_B)$  is represented as

$$G(P_A, P_B) = \begin{cases} S_e P_A \sqrt{\frac{\kappa}{R\theta_A} \left(\frac{2}{\kappa+1}\right)^{\frac{\kappa+1}{\kappa-1}}} & \left(\frac{P_B}{P_A} \leq b\right), \\ S_e P_A \sqrt{\frac{2\kappa}{(\kappa-1)} \frac{1}{R\theta_A} \left[ \left(\frac{P_B}{P_A}\right)^{\frac{2}{\kappa}} - \left(\frac{P_B}{P_A}\right)^{\frac{\kappa+1}{\kappa}} \right]} & \left(\frac{P_B}{P_A} > b\right), \end{cases} \quad (2)$$

where  $S_e$  is the effective area of the orifice,  $\kappa$  is the specific heat ratio,  $\theta_A$  is the upstream temperature, and  $b = 0.5283$  is the critical pressure ratio. In our simulation, the mass flow into the pipeline  $G_1$  is represented as

$$G_1 = \begin{cases} G(P_S, P_1) & (P_1 \leq P_S), \\ -G(P_1, P_S) & (P_1 > P_S). \end{cases} \quad (3)$$

The velocity of the air at the beginning of the pipeline is calculated by dividing the mass flow by the density of the air  $\rho_1$  and the inner diameter of the pipeline  $D$ , that is,

$$u_1 = \frac{G_1}{\pi \left(\frac{D}{2}\right)^2 \rho_1}. \quad (4)$$

### 3.1.2 Pipeline model

We implemented the distributed model of the pipeline.<sup>(22,23)</sup> The model is based on the following four equations for the air in the pipeline:

$$P = \rho R \theta, \quad (5)$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = -\frac{1}{\rho} \frac{\partial P}{\partial x} - \frac{\lambda}{2D} u^2, \quad (6)$$

$$\frac{\partial \rho}{\partial t} + \rho \frac{\partial u}{\partial x} + u \frac{\partial \rho}{\partial u} = 0, \quad (7)$$

$$\frac{\partial \theta}{\partial t} = -\frac{4h(\theta - \theta_a)}{\rho C_v D} - u \frac{\partial \theta}{\partial x} - \frac{R\theta}{C_v} \frac{\partial u}{\partial x} + \frac{1}{C_v} \lambda |u| u^2, \quad (8)$$

where  $P$  is the pressure at position  $x$  and time  $t$ ,  $\rho$  is the density,  $\theta$  is the temperature,  $u$  is the velocity of the air,  $\lambda$  is the coefficient of pipe friction,  $h$  is the heat transfer coefficient,  $\theta_a$  is the temperature of the atmosphere, and  $C_v$  is the specific heat of the gas at a constant volume. Equation (5) is the equation of state, Eq. (6) is the equation of motion, Eq. (7) is the equation of continuity, and Eq. (8) is the equation of energy conservation. In the model, the pipeline is spatially discretized using  $N$  grids with length  $\Delta x$ , as shown in Fig. 2(b). Using the sampling time  $\Delta t$ , we discretize Eqs. (5)–(8) as follows:

$$P_{i,j} = \rho_{i,j} R \theta_{i,j}, \quad (9)$$

$$u_{i,j+1} = u_{i,j} - \frac{\Delta t}{\Delta x} u_{conv} - \frac{1}{\bar{\rho}} \frac{\Delta t}{\Delta x} (P_{i,j} - P_{i-1,j}) - \frac{\lambda \Delta t}{2D} |u_{i,j}| u_{i,j}, \quad (10)$$

$$\rho_{i,j+1} = \rho_{i,j} - \frac{\Delta t}{\Delta x} \rho_{i,j} (u_{i+1,j} - u_{i,j}) - \frac{\Delta t}{\Delta x} \rho_{conv}, \quad (11)$$

$$\theta_{i,j+1} = \theta_{i,j} - \Delta t \frac{4h(\theta_{i,j} - \theta_a)}{\rho_{i,j} C_v D} - \frac{\Delta t}{\Delta x} \theta_{conv} - \frac{\Delta t}{\Delta x} \frac{R\theta_{i,j}}{C_v} (u_{i+1,j} - u_{i,j}) + \frac{\lambda \Delta t}{2C_v D} |\bar{u}_{i,j}|^3, \quad (12)$$

where  $P_{i,j}$ ,  $u_{i,j}$ ,  $\rho_{i,j}$ , and  $\theta_{i,j}$  are the variables in the  $i$ th block at the  $j$ th time step,  $\bar{\rho}_{i,j} = (\rho_{i-1,j} + \rho_{i,j})/2$ , and  $\bar{u}_{i,j} = (u_{i,j} + u_{i,j+1})/2$ . Convective terms  $u_{conv}$ ,  $\rho_{conv}$ , and  $\theta_{conv}$  are defined by upwind differences as follows:

$$u_{conv} = \frac{u_{i,j} + |u_{i,j}|}{2} (u_{i,j} - u_{i-1,j}) + \frac{u_{i,j} - |u_{i,j}|}{2} (u_{i+1,j} - u_{i,j}), \quad (13)$$

$$\rho_{conv} = \frac{u_{i,j} + |u_{i,j}|}{2} (\rho_{i,j} - \rho_{i-1,j}) + \frac{u_{i,j} - |u_{i,j}|}{2} (\rho_{i+1,j} - \rho_{i,j}), \quad (14)$$

$$\theta_{conv} = \frac{u_{i,j} + |u_{i,j}|}{2} (\theta_{i,j} - \theta_{i-1,j}) + \frac{u_{i,j} - |u_{i,j}|}{2} (\theta_{i+1,j} - \theta_{i,j}). \quad (15)$$

The coefficient of pipe friction  $\lambda$  is expressed as follows:

$$\lambda = \begin{cases} \frac{64}{Re} & (Re < 3000), \\ 0.3164 Re^{-0.25} & (Re \geq 3000), \end{cases} \quad (16)$$

where  $Re = \bar{\rho}|u_{i,j}|D/\mu$  is the Reynolds number and  $\mu$  is the viscosity of the air. The first equation is the coefficient of laminar flow and the second is the coefficient of turbulence (Blasius equation). The heat transfer coefficient  $h$  is calculated using the thermal conductivity of the air  $k$  as follows:

$$k = 7.95 \times 10^{-5} \theta_{i,j} + 2.0465 \times 10^{-3}, \quad (17)$$

$$h = \frac{0.046 Re^{0.8} Pr^{0.4} k}{D}, \quad (18)$$

where  $Pr = 0.72$  is the Prandtl number.

### 3.1.3 Chamber

Similarly to the input, we treat the flow into the final chamber as an orifice flow. The mass flow into the chamber  $G_C$  is represented as

$$G_c = \begin{cases} G(P_N, P_C) & (P_C \leq P_N), \\ -G(P_C, P_N) & (P_C > P_N), \end{cases} \quad (19)$$

where  $P_N$  is the pressure in the  $N$ th grid of the pipeline model and  $P_C$  is the pressure in the chamber.

The velocity of the air flow at the end of the pipeline is represented as

$$u_C = \frac{G_C}{\pi \left(\frac{D}{2}\right)^2 \rho_C}, \quad (20)$$

where  $\rho_C$  is the density of the air in the chamber.

The state change in the chamber is also assumed to be an isothermal process, that is,  $\theta_C = \text{const}$ . The mass of the air in the chamber is changed by the air flow and calculated as  $m_{C,j+1} = m_{C,j} + G_C \Delta t$ . Using the mass, we represent the density and pressure of the air as

$$\rho_C = \frac{m_C}{V_C}, \quad (21)$$

$$P_C = \rho_C R \theta_C, \quad (22)$$

where  $V_C$  is the volume of the chamber.

### 3.2 Experiment with the simulation model: NARMA model emulation

To assess the computational ability of the pipeline using various properties, we carried out an experiment using the simulation model to evaluate the performance of emulating a nonlinear system using the system with a single tube.

#### 3.2.1 Emulation task

The system used for emulation is a nonlinear autoregressive moving average (NARMA) model, which has been used as a benchmark in previous studies.<sup>(5,6)</sup> The following third-order NARMA model is used in this study:

$$y_i = 0.3y_{i-1} + 0.05y_{i-1} \sum_{k=0}^2 y_{i-k-1} + 1.5x_{i-2}x_i + 0.1, \quad (23)$$

where  $x_i$  is the input and  $y_i$  is the output of the NARMA model. In this study, the input  $x_i$  is uniformly distributed random values within the range  $[0, 0.2]$ .

The emulation task is to estimate the output of the system from the input sequentially. Figure 2(c) shows the experimental setting in the simulation. In the simulated setting, the input sequence  $x_i$  was transformed to a continuous signal  $x(t)$  by a zero-order hold with one sample interval  $T$ . The signal was transformed to pressure in the first chamber  $P_S$  by the linear equation  $P_S = a_0 + a_1 x$ . The pressures  $P_{m1}(t), \dots, P_{mK}(t)$  at  $K$  points located at intervals of  $L$  were measured at time intervals of  $T/M$ , which means that the pressure was measured  $M$  times for each input value  $x_i$ , following the time multiplexing technique.<sup>(6)</sup> For every  $M$  samples, the estimated output  $\hat{y}_i$  was calculated as



$$\hat{y}_i = w_0 + \sum_{k=1}^K \sum_{j=1}^M w_{kj} P_{mk} \left( iT + j \frac{T}{M} \right), \quad (24)$$

where  $w_0, w_{11}, \dots, w_{KM}$  are the weights for the estimation. The weights were optimized using training data by ridge regression as follows:

$$\mathbf{w}_o = \underset{[w_0, w_{11}, \dots, w_{KM}]}{\operatorname{arg\,min}} \left( \sum_{i=1}^{N_{train}} (y_i - \hat{y}_i)^2 + \alpha \left( w_0^2 + \sum_{k=1}^K \sum_{j=1}^M w_{kj}^2 \right) \right), \quad (25)$$

where  $\mathbf{w}_o$  is the vector of optimized weights,  $N_{train}$  is the number of samples in the training data, and  $\alpha$  is a positive constant used as a penalty of parameters, which prevents extreme parameters caused by overfitting. Note that the minimization problem has an analytical solution and can be solved efficiently.<sup>(24)</sup>

The performance of the task can be evaluated by measuring the error between the estimated output and the original output. In this study, we calculated the root-mean-square error (RMSE) as follows:

$$RMSE = \sqrt{\frac{1}{N_s} \sum_{i=0}^{N_{test}} (y_i - \hat{y}_i)^2}, \quad (26)$$

where  $N_{test}$  is the number of samples used for the evaluation.

### 3.2.2 Experiment

We evaluated the performance of the NARMA emulation task while varying the inner diameter of the pipeline and the pipeline length. The fixed parameters are shown in Table 1. The

Table 1  
Parameters in the simulation experiment.

Parameter name	Value
Number of sensors $K$	4
Number of samples for estimation $M$	5
Input interval $T$	$1 \times 10^{-1}$ s
Sampling time $\Delta t$	$5 \times 10^{-7}$ s
Temperature of atmosphere $\theta_a$	293 K
Input	
Pressure intercept $a_0$	195 kPa
Pressure coefficient $a_1$	50 kPa
Temperature $\theta_s$	293 K
Effective orifice area $S_e$	$1 \times 10^{-6}$ m <sup>2</sup>
Pipeline	
Length of each grid $\Delta x$	$2 \times 10^{-1}$ m
Chamber	
Volume of chamber $V_C$	$6 \times 10^{-5}$ m <sup>3</sup>
Effective orifice area $S_e$	$1 \times 10^{-6}$ m <sup>2</sup>

chamber resembles a pneumatic artificial rubber muscle (PARM), which is a soft actuator and used in the experiment in Sect. 4. The pressure range was set to be around the pressure that can lead to a detectable pressure change caused by a wearer's motion in our study of a gait assistive suit using the PARMs.<sup>(25)</sup> The other parameters were selected to be realistic values in the use of PARMs.

Eightfold cross-validation was performed. Eight sets of data for training and testing were generated. For each set, 200 samples of the input sequence  $x_i$  and the corresponding output sequence  $y_i$  were calculated. The input sequence was used to generate 20 s of pressure values. Using seven of the eight sets, we optimized the weights for the estimation. Using the other set as the test set, we estimated the output sequence from the pressure, and RMSE was calculated. This process was repeated for every selection of the test set.

We evaluated the estimation performance for every combination of the inner diameter [ $D = 1$  to 2.5 (step 0.25), 3, 3.5, 4 mm] and the interval between sensors ( $L = 0.6, 1.0, 1.4, 1.8$  m, which correspond to the pipeline lengths of 3, 5, 7, and 9 m, respectively). The mean RMSE values for the different combinations of conditions were compared. To prevent the effect of small oscillations in the pressure values caused by numerical errors, we added normally distributed noise with a standard deviation of 0.05 kPa to the pressure values before the training. For the ridge regression,  $\alpha$  was set to  $10^{-3}$ .

### 3.2.3 Results

The mean RMSE for each combination of conditions is shown in Fig. 3. For each length, the relationship between the diameter and RMSE is shown. The horizontal dotted line shows the error of estimation by linear regression using the input sequence  $x_i$  ( $\hat{y}_i = w_0 + w_1 x_i$ , where  $w_0$  and

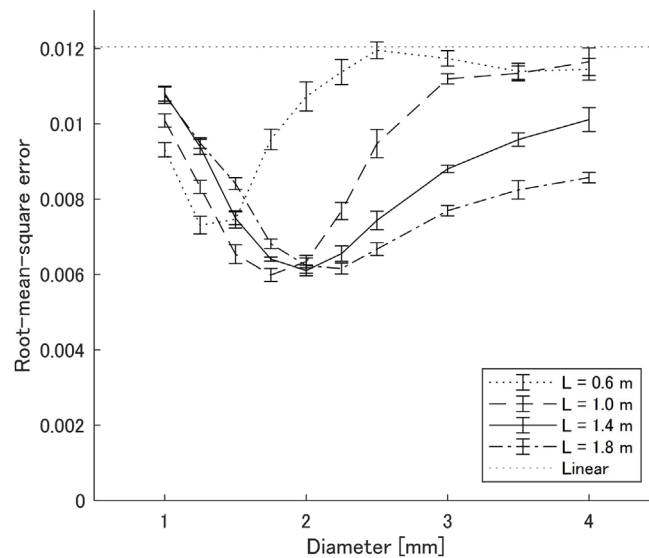


Fig. 3. Performance of NARMA model emulation task (simulation). The mean RMSE values were calculated by eightfold cross-validation. Error bars indicate standard errors.

$w_1$  were optimized by ridge regression with  $\alpha = 10^{-3}$ ). In the cases of  $L = 1.0, 1.4,$  and  $1.8$  m, the RMSE was the smallest around  $D = 2$  mm.

Figure 4 shows examples of estimation results and pressure changes for  $D = 1, 2,$  and  $4$  mm with  $L = 1.4$  m. The result of estimation by the linear regression is also shown. In the case of  $D = 1$  mm, the estimation result was more similar to the result of linear regression than the target output. Pressure values in the pipeline started to approach the input pressure within around  $0.03$  s after the transition of the input, then the rate of change decreased until the next transition of the input. The pressure values during the last  $0.07$  s of each input value were almost equally distributed between the input pressure and the pressure in the chamber. In the case of  $D = 2$  mm, the estimated results were similar to the target. The pressure values underwent complicated changes. After the transition of the input, the pressure in the pipeline immediately changed toward the input pressure. At around  $0.03$  s after the transition, the pressure changed toward the previous pressure and changed slowly toward the input pressure again. The timing of the two changes depended on the position in the pipeline. The pressure in the pipeline changed faster than in the case of  $D = 1$  mm, but in most cases, the pressure did not reach the input pressure within  $T = 0.1$  s. The final pressure values corresponding to the input, which were the initial pressure values corresponding to the next input, changed the subsequent pressure trajectories extensively. This means that the effect of an input was also propagated to the results of the following inputs. In the case of  $D = 4$  mm, the estimated value was not similar to the target value. The trajectories of pressure values in the pipeline were almost the same among the locations. The pressure values slowly changed after the transition of the input, and in most cases reached the input pressure at the end of a single input. Therefore, the pressure values generated by an input hardly affected the results of subsequent inputs.

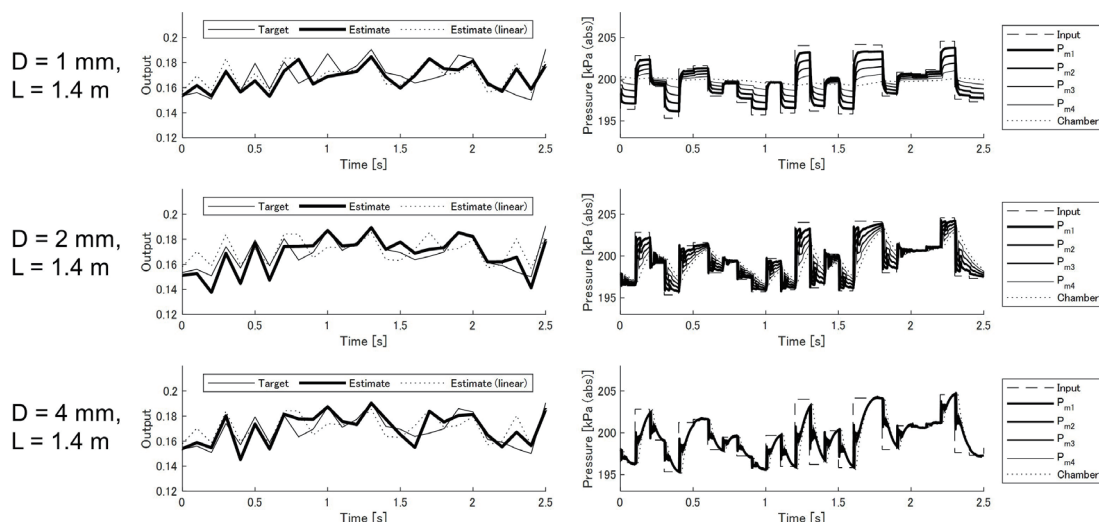


Fig. 4. Examples of results of NARMA model emulation task for different diameters (simulation). (Left) Target and estimated values. (Right) Pressure in the tube used for the estimation. In the right panels, noise added in the evaluation is not shown.

Figure 5 shows results for different pipeline lengths. In the case of  $D = 2$  mm, changing the pipeline length affected the response speed. As shown in Fig. 5(a), shortening the pipeline made the response reach the input pressure. In this case, the estimation performance was worse than that for other lengths (Fig. 3). In the case of  $D = 3$  mm, changing the pipeline length changed the trajectory of pressure. When the pipeline was short, the pressure responses were almost the same among the locations. When the pipeline was long, the trajectory in the beginning of response varied among the locations. In this case, lengthening the pipeline improved the estimation performance (Fig. 3).

### 3.3 Step response

To present how the dynamics of the pressure are regulated according to the parameters of pipelines more clearly, we simulated step responses of the pressure. Pressure in the simulated instruments was set to 200 kPa and pressure in the first chamber was instantly changed to 205 kPa. These pressures corresponded to the central and maximum values of the input of the NARMA emulation task. The responses were simulated for the combination of the diameter  $D = 1, 2, 3,$  and 4 mm and the length  $L = 0.6, 1, 1.4,$  and 1.8 m.

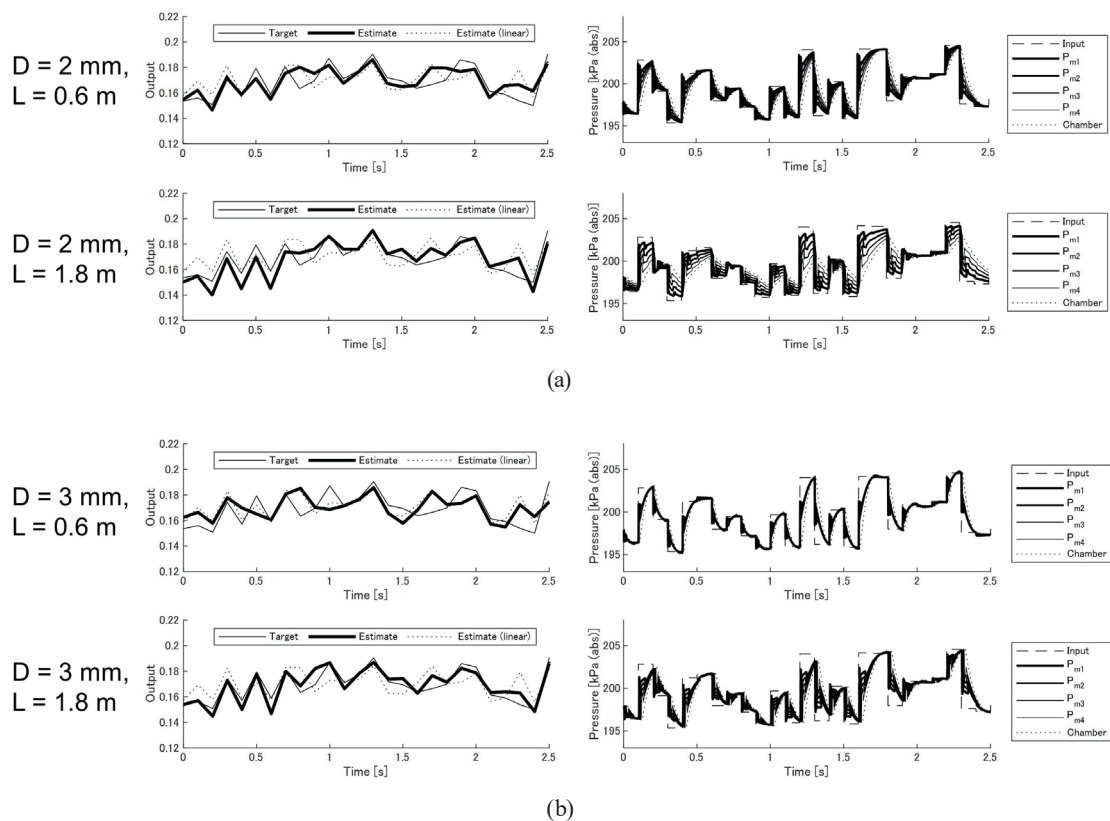


Fig. 5. Examples of results of NARMA model emulation task for different tube lengths (simulation) with (a)  $D = 2$  mm and (b)  $D = 3$  mm. (Left) Target and estimated values. (Right) Pressure in the tube used for the estimation. In the right panels, noise added in the evaluation is not shown.

Figure 6 shows the simulated step responses in all combinations. As the diameter was increased, the pressure response was changed from the step-like trajectory to smooth trajectory with initial step-like or complicated response, and the variability of pressure values between sensors was decreased. When the length of the pipeline was increased, the onsets of responses were delayed in proportion to the distance between the sensor and the inlet. This increased the variability of pressure values between sensors in the beginning of response. In the cases of  $D = 2$  mm, as the length was increased, the responses following the complicated trajectory were slowed down, and the variety of sensor values at the time of input interval  $T$  (vertical dotted lines in Fig. 6) was increased. In the case of  $D = 1$  mm, the response in the chamber slowed down as the length was increased and approached a constant value, and the sensor values at the time of input interval  $T$  were almost equally distributed between the input and the original pressure.

### 3.4 Memory capacity

To evaluate the performance of fading memory of the pipeline quantitatively, we calculated an index of memory of the pipeline with various parameters. The index, memory capacity,<sup>(26)</sup> is defined on the basis of the result of the task to retrieve past input from reservoir response. We calculated the memory capacity for each combination of conditions in the NARMA emulation task. First, the weights for estimation were optimized to estimate the  $k$ -sample delayed input sequence, i.e.,  $y_i = x_{i-k}$ . Then,  $k$ -delay memory capacity was calculated as follows:

$$MC_k = \frac{\text{cov}^2(x_{i-k}, \hat{y}_i)}{\sigma^2(x_i)\sigma^2(\hat{y}_i)}, \quad (27)$$

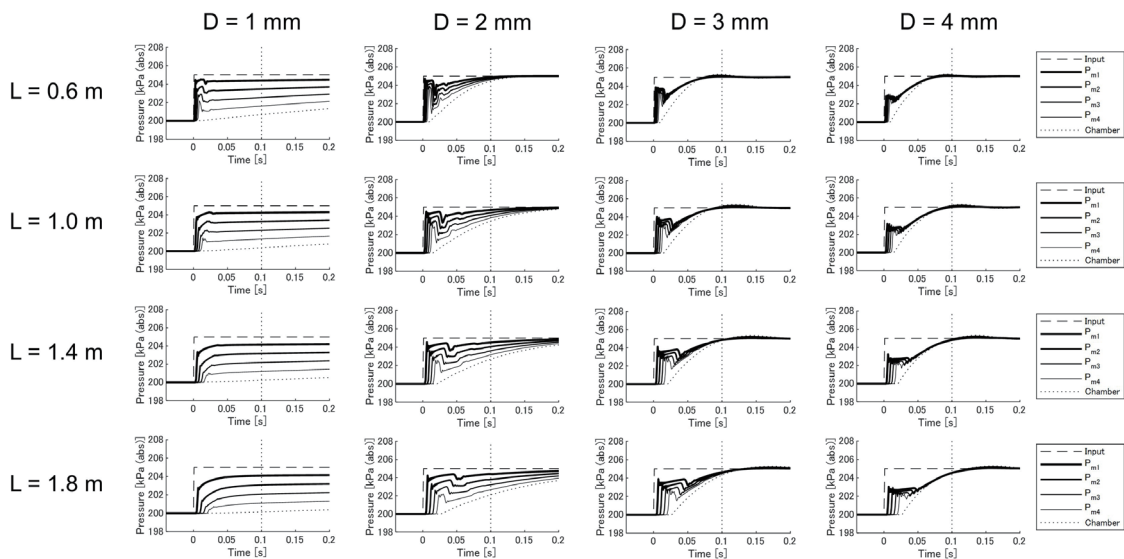


Fig. 6. Step responses of pressure in the pipeline (simulation). Vertical dotted lines show the time length of input interval ( $T = 0.1$  s) in the NARMA emulation task.

where  $cov(x, y)$  was the covariance between  $x$  and  $y$ , and  $\sigma^2(x)$  was the variance of  $x$ . The memory capacity was calculated as follows:

$$MC = \sum_{k=1}^{N_{max}} MC_k, \quad (28)$$

where  $N_{max}$  was the maximum delay we considered, which was 20 samples in this study.

Figure 7(a) shows the memory capacity under the conditions of diameter and length. The maximum memory capacity was marked in the cases around  $D = 2$  mm and  $L \geq 1.0$  m. The results show that longer tubes have larger memory capacity when  $D \geq 2$  mm; however, the capacity hit the ceiling at around 3. Figure 7(b) shows the  $k$ -delay memory capacity under the conditions with  $D = 1, 2, 3, 4$  mm. Most conditions have high 1-delay memory capacity, and the difference between the conditions was mainly caused by the amount of 2-delay memory capacity. When the delay was larger than 2, the  $k$ -delay memory capacity rapidly decreased in the cases with  $D \geq 2$  mm. In the cases with  $D = 1$  mm, the  $k$ -delay memory capacity with  $k \geq 2$  had a long-tailed distribution, and its amount was larger in short tubes.

### 3.5 Discussion

For the experimental settings in the simulation, we found that the diameter and length affected the computation performance. In the cases that we evaluated, parameters around  $D =$

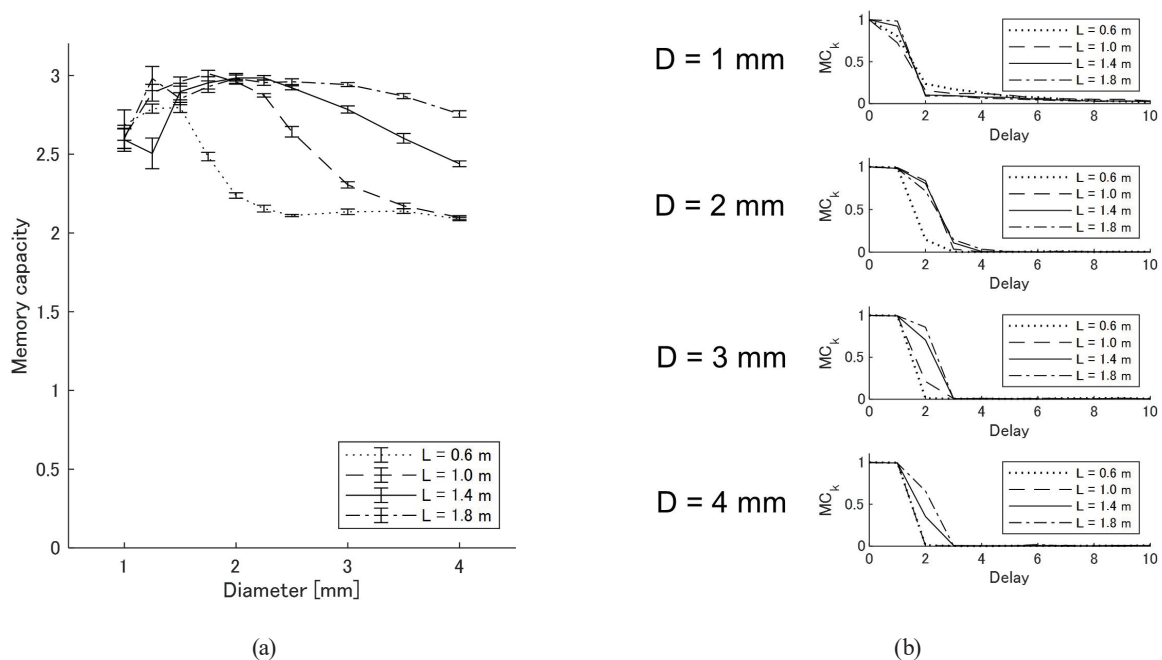


Fig. 7. Memory capacity of the simulated pipeline. (a) Memory capacity under the conditions of the NARMA model emulation. The mean memory capacities were calculated by eightfold cross-validation. Error bars indicate standard errors. (b)  $k$ -delay memory capacity ( $MC_k$ ). The mean values by eightfold cross-validation are plotted.

2 mm and  $L \geq 1.0$  m were optimal in terms of RMSE. The first effect of the parameters is on the memory function. As shown in Fig. 4, in the case with the optimal diameter, pressure values in a single step include the information of current and previous inputs as the distribution of pressure between sensors and time points. In addition, transient pressures at some locations at the time of updating the input provides information about the input two steps earlier for the response to the new input. As the output of the third-order NARMA system depends on the input two steps earlier, the information was used effectively in the task. When the diameter is large, as shown in Fig. 6, the flow in the pipeline is large, and the pressure in the pipeline and chamber quickly reach the input pressure. This prevents the propagation of the input two steps earlier; however, lengthening the pipeline makes the transient state longer and increases its memory capacity. Additionally, as shown in the beginning of step responses (Fig. 6), the length of the pipeline also enhances the function as a delay line, and makes the pipeline preserve the previous state directly. Thus, with the adequate diameter and length of the tube, the delay of pressure propagation leads to the memory of past inputs. The effect of past inputs gradually disappears as a result of friction, thus providing the fading memory required for reservoir computing.<sup>(9,13)</sup> When the diameter is small, as the case with  $D = 1$  mm, the pressure propagation in the tube is extremely slowed down. In these cases, the pressure at the end of the pipeline hardly changes, and the distribution of pressure in the pipeline is mostly determined by only the current input pressure. This decreases the memory capacity, and leads to a peak of the memory capacity around a given diameter, which would correspond to the peak in RMSE. However, it leads to a weak but long-term memory of the information on past inputs, as shown in Fig. 7(b). This effect decreases with long tubes because the pressure propagation is almost halted. The effect of long-term memory might be advantageous in tasks that require the memory of distant past inputs.

The second effect of the parameters is to provide various mappings from the input. For the optimal diameter and length, the response of the pressure in the pipeline has a complicated shape. The response has different pressures at different locations and time points, resulting in high dimensionality. Additionally, the pipeline as a delay line makes the spatially distributed sensors have similar function as time multiplexing, which also leads to high-dimensional mapping. On the other hand, the response is nearly constant at each location in the case of a small diameter. Besides, the response in the case of a large diameter and short length is similar at all locations. These phenomena will decrease the effective dimension of the pressure values. Nonlinearity of the mapping would contribute to the result of the NARMA emulation task because the emulated system requires nonlinear computation; however, the effect of nonlinearity should be investigated more extensively by using indices that deal with nonlinear effects, e.g., information processing capacity.<sup>(27)</sup>

The memory capacities and mapping in the pipeline will also be affected by the number of sensors  $K$  and time multiplexing  $M$ . Increasing the number of sensors will be beneficial to increase the dimension of signals without increasing the number of time multiplexing. This will affect the high-dimensional mapping implemented by the reservoir. These effects are required to be investigated for the design of the pneumatic reservoir in future works.

#### 4. Application to a Soft Exoskeleton

To show the proposed method in a real application, we applied it to the posture estimation of a soft exoskeleton. We have already developed a soft exoskeleton for lower limbs using PARMs.<sup>(25)</sup> In a previous study, we used some of the PARMs as sensors to detect the timing in the gait phase to apply an assisting force to the wearer.<sup>(25)</sup> In the present study, we used PARMs to continuously estimate the joint angle toward achieving precise assistance. We supposed that the air connection of PARMs for the estimation was separated from the PARMs for actuation as in the previous study.<sup>(25)</sup> In the following experiment, we used the PARMs only for estimation.

As shown in Figs. 8(a) and 8(b), two PARMs on the thighs of the wearer are used for the estimation. Note that the PARMs are not used as actuators but as chambers. The two PARMs are

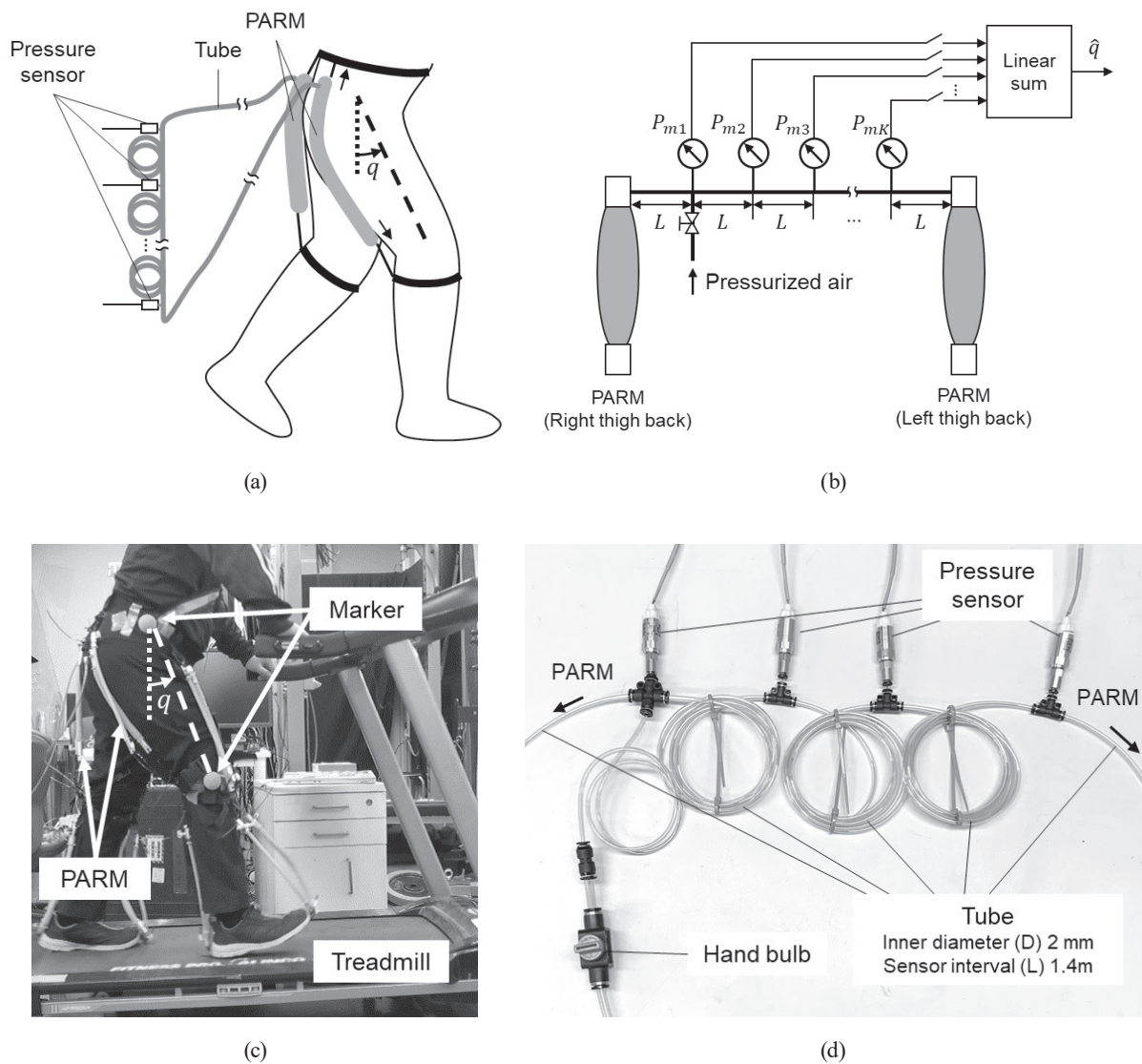


Fig. 8. Posture estimation of a soft exoskeleton using the proposed method. (a) Overview of hardware configuration, (b) estimation system, (c) experimental setup, and (d) example of tube connection (inner diameter 2 mm, sensor interval 1.4 m).



connected by a tube, and pressure sensors are placed on the tube. Before their use, the internal spaces of the pipeline and the PARMs are filled with pressurized air and sealed by a hand bulb on the supply port. When the wearer moves his/her right leg forward, the PARM on the right thigh is elongated and pushes the internal air into the tube. This changes the pressure in the tube, and the pressure is measured by the sensors. The measured pressure signals are used to estimate the thigh angle  $q(t)$  using the following equation:

$$\hat{q}(t) = w_0 + \sum_{k=1}^K \sum_{j=1}^M w_{kj} P_{mk} \left( t - T + j \frac{T}{M} \right), \quad (29)$$

where  $w_0, w_{11}, \dots, w_{KM}$  are the weights used for the estimation, which are optimized using ridge regression before the estimation,  $K$  is the number of pressure sensors,  $M$  is the number of samples used for the estimation,  $P_{mk}(t)$  is the pressure measured by the  $k$ th sensor at time  $t$ , and  $T$  is the width of the time window for the samples used in the estimation.

We examined the feasibility of the method and the effect of the tube parameters in an experiment. Figure 8(c) shows the experiment setup. We used a soft exoskeleton with two PARMs<sup>(25)</sup> (Bridgestone, length 0.3 m, inner diameter 9.5 mm) on the back of the left and right thighs in this study. The two PARMs were connected to the back of the waist and the top of the calves. A tube (Pisco, UBT series, polyurethane) was connected to the PARMs, as shown in Fig. 8(d), and four pressure sensors (SMC, PSE530) were attached to the tube at predetermined intervals using fittings (Pisco, Mini series). We prepared three types of tube connection with inner diameter  $D = 2$  mm and sensor interval  $L = 1.4$  m, with  $D = 4$  mm and  $L = 1.4$  m, and with  $D = 2$  mm and  $L = 0.6$  m. The sensor values were measured using a data acquisition device (National Instruments, NI USB-6211) and software (National Instruments, DAQExpress) with a sampling frequency of 31.25 kHz. To measure the angle of the right thigh, which was the value to be estimated, ping-pong balls were attached on the side of the waist and knee of the soft exoskeleton. The balls were used to reconstruct the thigh angle from the recorded video (frame rate 29.97 fps) by an in-house program using a computer vision library (OpenCV). A motorized treadmill (Alinco, AFW1014) was used to measure the gait movement in a restricted space. The thigh angle and pressure signals were resampled with a sampling frequency of 50 Hz. The parameters used in the estimation are shown in Table 2.

In the experiment, an able-bodied participant (male, 36 years old) wore the soft exoskeleton. The PARMs and tube were filled with pressurized air to about 100 kPa(G), for which it has been confirmed that gait timing can be detected.<sup>(25)</sup> The wearer walked on the treadmill for 28 s at four different speeds (1, 2, 3, and 4 km/h). The pressure in the tube and the video for measuring

Table 2  
Parameters in the posture estimation experiment.

Parameter name	Value
Number of sensors $K$	4
Number of samples for estimation $M$	5
Width of time window $T$	$1 \times 10^{-1}$ s

the right thigh angle were recorded. The experiment was performed with  $D = 2$  mm and  $L = 1.4$  m,  $D = 4$  mm and  $L = 1.4$  m, and  $D = 2$  mm and  $L = 0.6$  m.

In the evaluation, fourfold cross-validation was performed. The data for each walking speed were divided into seven 4 s blocks. We picked up the first, third, fifth, and seventh blocks and constructed four data sets that include data for all speeds. Using three of the four sets as training data, we optimized the weights for the estimation. Again, ridge regression with  $\alpha = 10^{-3}$  was used. Using the other set as the test set, we estimated the output sequence from the pressure, and the RMSE was calculated for each combination of tube properties and speed. Note that the weights were commonly used for all speeds, but the error in the test data was evaluated for each speed because we observed that the error depended on the speed. This process was repeated for every selection of the test set.

Figure 9 shows the mean RMSE of the thigh angle estimation. For the speeds of 2, 3, and 4 km/h, the tube with  $D = 2$  mm and  $L = 1.4$  m showed a smaller error than the other tubes, consistent with the simulation results. For the speed of 1 km/h, all tubes showed a similar RMSE.

Figure 10 shows examples of the estimation results. The target thigh angle, the estimated angle [Fig. 10(a)], and the pressure signals used for the estimation [Fig. 10(b)] are shown. The tubes with  $D = 4$  mm gave a larger estimation error than the tubes with  $D = 2$  mm. This was caused by the decreased dimension of the pressure signals with  $D = 4$  mm, shown in Fig. 10(b). The difference of dimensions was more clearly shown by principal component analysis.<sup>(28)</sup> For each walking speed, the pressure signals were decomposed into principal components, and the ratio of each component's variance to the total variance, which was referred to as the variance

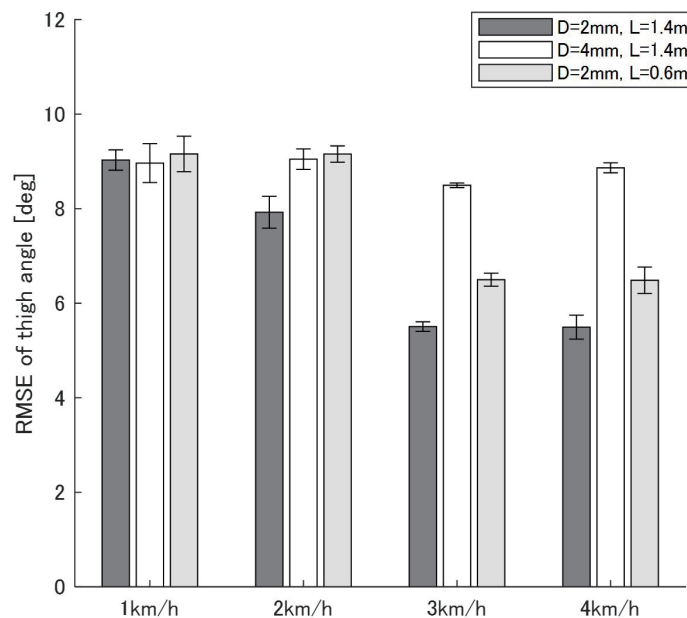


Fig. 9. Performance of the posture estimation. The mean RMSE was calculated by fourfold cross-validation. Error bars indicate standard errors.

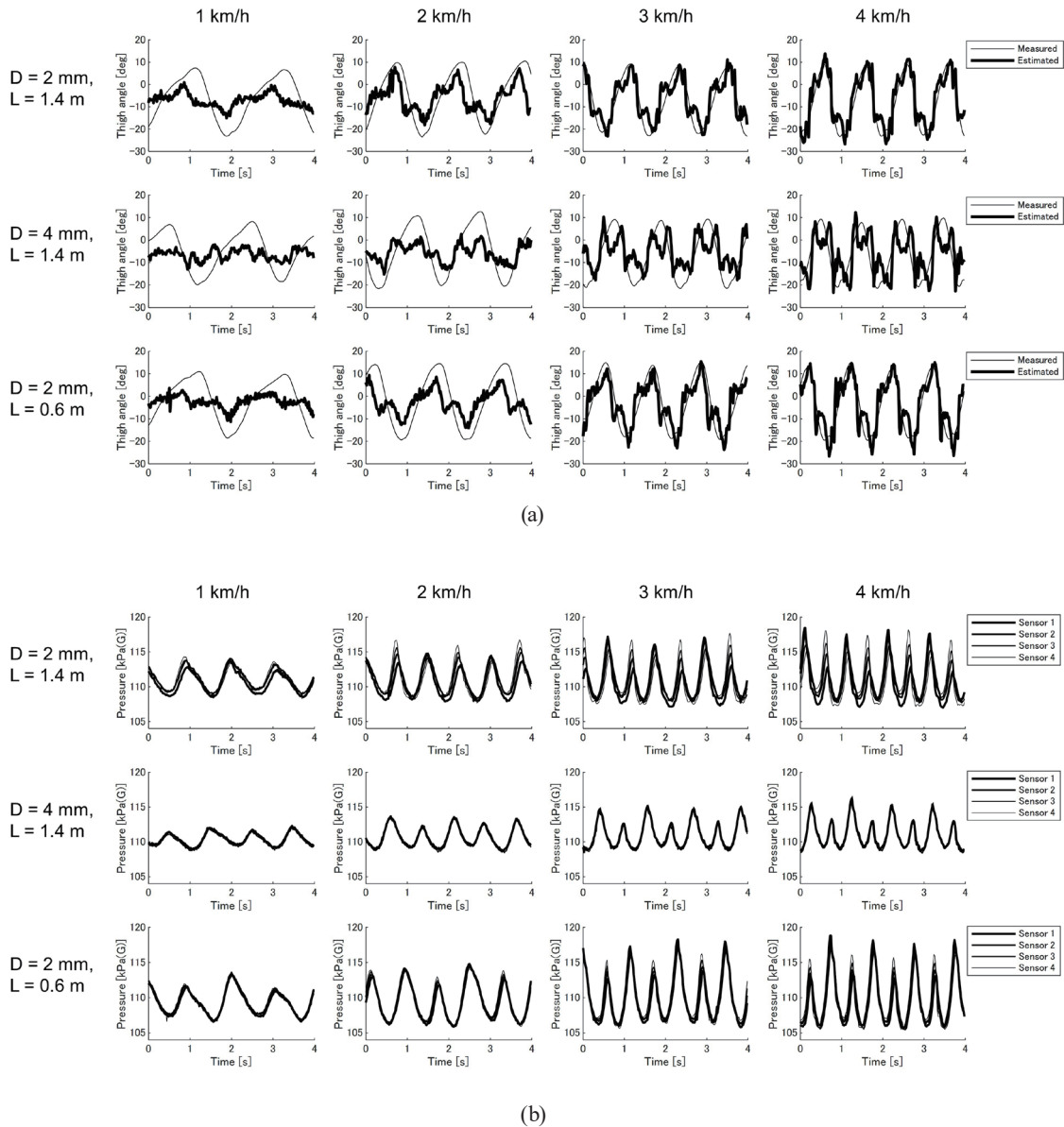


Fig. 10. Examples of the results of posture estimation. (a) Target and estimated values. (b) Pressure in the tube used for the estimation.

explained, was calculated. Figure 11(a) shows the results. The variances were concentrated to the first two principal components under all conditions, and the variance of the second principal component was smallest when  $D = 4$  mm, which shows that the effective dimension of the pressure signals was smallest in this case. The analysis was also applied to  $KM$ -dimension signals for the estimation, i.e.,  $[P_{m1}(t-T/M), P_{m1}(t-2T/M), \dots, P_{mK}(t-(M-1)T/M), P_{mK}(t-T)]$ . The results of analysis also showed a similar tendency to the former one [Fig. 11(a)]. The error in the estimation for the speed of 1 km/h was high under all conditions. As shown in Fig. 10(b), decreased dimension of the pressure signals was also found in this case, and this phenomenon

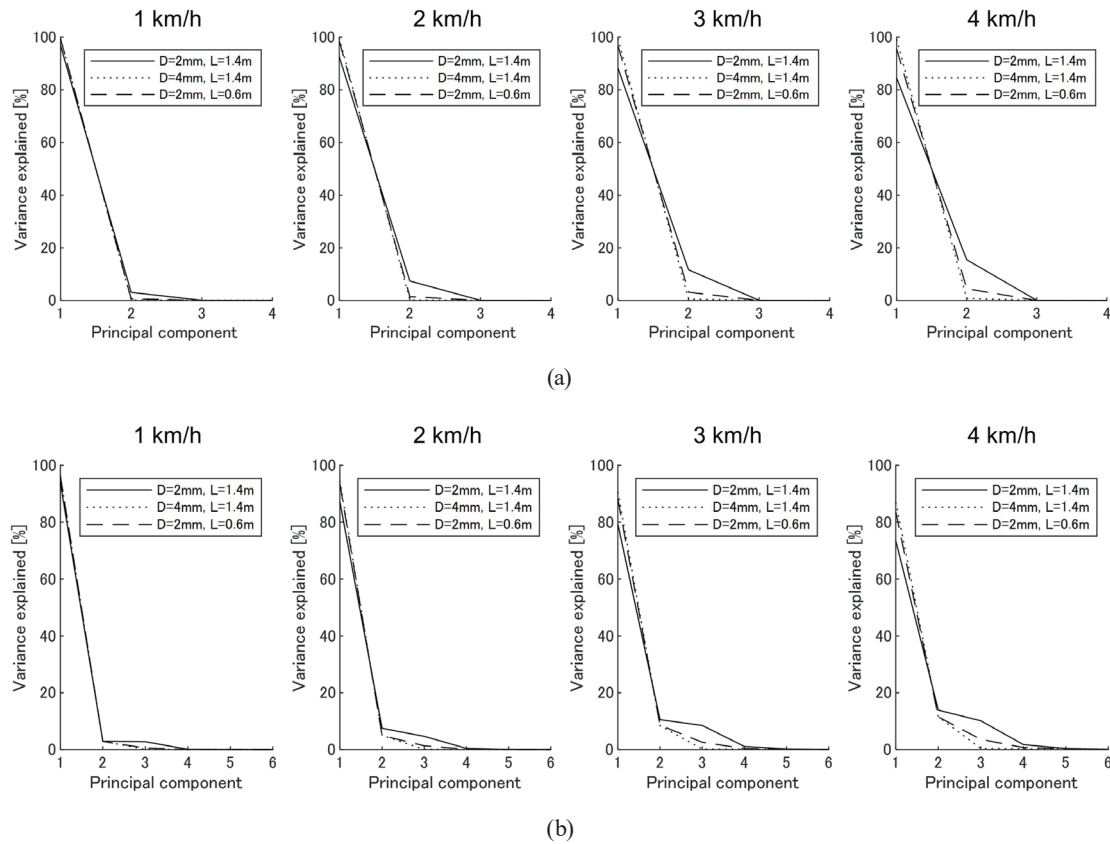


Fig. 11. Variance of principal components of the pressure. (a) Original  $K$  (4)-dimensional sensor signals. (b)  $KM$  (20)-dimensional signals that were used for the posture estimation. The variances of only the first six principal components are shown.

was also shown by the small variance of the second principal component in Figs. 11(a) and 11(b). This was caused by the frequency characteristic of the air in the pipeline, which passes the low-frequency pressure with little phase delay.<sup>(29)</sup> For actual application in soft robotics, the computational ability of the method should be assessed from the viewpoint of frequency characteristics, which will be our future work.

The estimated angles had the error of  $5.5^\circ$  in the best cases (Fig. 9,  $D = 2$  mm,  $L = 1.4$  m, speed of 3 or 4 km/h). The error is comparable to the angle estimated by soft sensors for an exosuit under high running speed (e.g., nearly  $5^\circ$  for hip angle with a running speed of 2.7 m/s).<sup>(30)</sup> The estimated angles are not adequate for precise angle control; however, they would be usable for the state estimation of gait, e.g., the estimation of gait mode (walking or running) or gait phases (initial contact, loading response, mid-stance, terminal stance, pre-swing, initial swing, mid-swing, and terminal swing),<sup>(31)</sup> which is required for selecting the adequate control mode. Our previous study of the method using the PARMs detected only the beginning of the pre-swing phase, recognized the time of a gait cycle, and sent predefined cyclic control signals.<sup>(25)</sup> The continuous estimation of the angle in the present study would contribute to generate control signals, which is adaptive to the change of the gate phase period during walking and assist the

wearer's body in more adequate timings. The present results are in the simplest case of the pneumatic reservoir computing, which has only one tube connection. Future work will be required for improving the results using extended instruments with multiple chambers and tube connections.

## 5. Conclusion

In this paper, we proposed a framework that can be used for the state estimation of soft robots by exploiting the dynamics of air in tubes connecting chambers in the robots. We focused on a case with a single tube, and showed that the framework can be used for a benchmark task and that the inner diameter and length of the tube affect the computational ability. We implemented a physical system based on the framework and applied it to posture estimation of a soft exoskeleton. The results showed that the system with a suitable diameter and length of the tube can be used for this purpose. Future issues include investigating the effect of parameters of input (e.g., number of sensors and signal frequency) on the estimation accuracy, and the performance of the method with more chambers and tube connections.

## Acknowledgments

This work was supported by JSPS KAKENHI Grant Numbers 20K04375 and 21H04544. Part of this research is based on the Cooperative Research Project of the Research Center for Biomedical Engineering.

## References

- 1 D. Rus and M. T. Tolley: *Nature* **512** (2015) 467. <https://doi.org/10.1038/nature14543>
- 2 R. Pfeifer, F. Iida, and G. Gómez: *Int. Congr.* **1291** (2006) 22. <https://doi.org/10.1016/j.ics.2005.12.080>
- 3 Q. Zhao, K. Nakajima, H. Sumioka, H. Hauser, and R. Pfeifer: *Proc. 2013 IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)* (2013) 1445. <https://doi.org/10.1109/IROS.2013.6696539>
- 4 K. Caluwaerts, M. D'Haene, D. Verstraeten, and B. Schrauwen: *Artif. Life* **19** (2013) 35. [https://doi.org/10.1162/ARTL\\_a\\_00080](https://doi.org/10.1162/ARTL_a_00080)
- 5 K. Nakajima, H. Hauser, T. Li, and R. Pfeifer: *Sci. Rep.* **5** (2015) 10487. <https://doi.org/10.1038/srep10487>
- 6 K. Nakajima, H. Hauser, T. Li, and R. Pfeifer: *Soft Robotics* **5** (2018) 339. <https://doi.org/10.1089/soro.2017.0075>
- 7 C. Fernando and S. Sojakka: *Lecture Notes in Computer Science* **2801** (2003) 588. [https://doi.org/10.1007/978-3-540-39432-7\\_63](https://doi.org/10.1007/978-3-540-39432-7_63)
- 8 K. Goto, K. Nakajima, and H. Notsu: *arXiv* (2020). <https://arxiv.org/abs/2001.08502>
- 9 G. Tanaka, T. Yamane, J. B. Héroux, R. Nakane, N. Kanazawa, S. Takeda, H. Numata, D. Nakano, and A. Hirose: *Neural Netw.* **115** (2019) 100. <https://doi.org/10.1016/j.neunet.2019.03.005>
- 10 M. Eder, F. Hisch, and H. Hauser: *Adv. Robotics* **32** (2018) 375. <https://doi.org/10.1080/01691864.2017.1402703>
- 11 E. Torres, K. Nakajima, and I. S. Godage: *Proc. 2019 IEEE Int. Conf. Soft Robotics (RoboSoft)* (2019) 441. <https://doi.org/10.1109/ROBOSOFT.2019.8722777>
- 12 H. Jaeger and H. Haas: *Science* **304** (2004) 78. <https://doi.org/10.1126/science.1091277>
- 13 W. Maass, T. Natschläger, and H. Markram: *Neural Comput.* **14** (2002) 2531. <https://doi.org/10.1162/089976602760407955>
- 14 H. Hauser, A. J. Ijspeert, R. M. Fuchslin, R. Pfeifer, and W. Maass: *Biol. Cybern.* **105** (2011) 355. <https://doi.org/10.1007/s00422-012-0471-0>
- 15 H. Hauser, A. J. Ijspeert, R. M. Fuchslin, R. Pfeifer, and W. Maass: *Biol. Cybern.* **106** (2012) 595. <https://doi.org/10.1007/s00422-012-0516-4>
- 16 L. Appeltant, M. C. Soriano, G. Van der Sande, J. Danckaert, S. Massar, J. Dambre, B. Schrauwen, C. R. Mirasso, and I. Fischer: *Nat. Commun.* **2** (2011) 468. <https://doi.org/10.1038/ncomms1476>

- 17 K. Vandoorne, P. Mechet, T. Van Vaerenbergh, M. Fiers, G. Morthier, D. Verstraeten, B. Schrauwen, J. Dambre, and P. Bienstman: *Nat. Commun.* **5** (2014) 3541. <https://doi.org/10.1038/ncomms4541>
- 18 C. Du, F. Cai, M. A. Zidan, W. Ma, S. H. Lee, and W. D. Lu: *Nat. Commun.* **8** (2017) 2204. <https://doi.org/10.1038/s41467-017-02337-y>
- 19 J. Torrejon, M. Riou, F. A. Araujo, S. Tsunegi, G. Khalsa, D. Querlioz, P. Bortolotti, V. Cros, K. Yakushiji, A. Fukushima, H. Kubota, S. Yuasa, M. D. Stiles, and J. Grollier: *Nature* **547** (2017) 428. <https://doi.org/10.1038/nature23011>
- 20 K. Fujii and K. Nakajima: *Phys. Rev. Appl.* **8** (2017) 24030. <https://doi.org/10.1103/PhysRevApplied.8.024030>
- 21 Y. Nakayama: *Introduction to Fluid Mechanics* (Butterworth-Heinemann, 2018) 2nd ed.
- 22 J. Li, K. Kawashima, T. Fujita, and T. Kagawa: *Precision Eng.* **37** (2013) 880. <https://doi.org/10.1016/j.precisioneng.2013.05.006>
- 23 D. Sakamoto, C. Youn, and T. Kagawa: *Adv. Mechanical Eng.* **5** (2015) 257283. <https://doi.org/10.1155/2013/257283>
- 24 G. H. Golub and C. F. Van Loan: *Matrix Computation* (Johns Hopkins Univ Pr, 2012) 4th ed.
- 25 T. Miyazaki, T. Tagami, D. Morisaki, R. Miyazaki, T. Kawase, T. Kanno, and K. Kawashima: *Appl. Sci.* **9** (2019) 2869. <https://doi.org/10.3390/app9142869>
- 26 H. Jaeger: GMD-Report **152** (2001). <http://publica.fraunhofer.de/documents/B-73131.html>
- 27 J. Dambre, D. Verstraete, B. Schrauwen, and S. Massar: *Sci. Rep.* **2** (2012) 514. <https://doi.org/10.1038/srep00514>
- 28 L. F. Abbott, K. Rajan, and H. Sompolinsky: *The Dynamic Brain: An Exploration of Neuronal Variability and Its Functional Significance*, Eds. M. Ding and D. L. Glanzman (Oxford University Press, New York, 2011) Chap. 4. <https://doi.org/10.1093/acprof:oso/9780195393798.003.0004>
- 29 T. Kagawa, A. Kitagawa, K. Sanada, and T. Takenaka: *Trans. Soc. Inst. Cont. Eng.* **21** (1985) 953 (in Japanese). <https://doi.org/10.9746/sicetri1965.21.953>
- 30 Y. Mengüç, Y.-L. Park, H. Pei, D. Vogt, P. M. Aubin, E. Winchell, L. Fluke, L. Stirling, R. J. Wood, and C. J. Walsh: *Int. J. Robot. Res.* **33** (2014) 1748. <https://doi.org/10.1177/0278364914543793>
- 31 J. Perry and J. Burnfield: *Gait Analysis: Normal and Pathological Function* (Slack Inc, Thorofare, 2010) 2nd ed.